

Architecture of an Open-Source Real-Time Distributed Cyber Physical System

C

Stefano Scanzio
CNR-IEIT, Italy

INTRODUCTION

Cyber physical systems (CPSs) (Guan et al., 2016) are based on a number of nodes connected through a communication network, which can interact with the environment. Application contexts that foresee the use of CPSs are Industry 4.0, building automation and smart grids (Yu & Xue, 2016). In such systems, nodes are connected through a wired or a wireless network, but commonly a typical configuration relies on the combination of both. In many applications, such as those regarding manufacturing systems, determinism on data exchanges between nodes of the system is the most important requirement.

On the other hand, in recent years, open-source software and applications are approaching CPS market. Open-source provides a low cost and effective method to implement inexpensive, performing and bug free applications. The need of open-source components for many application contexts derives from their capability to be highly customizable. This characteristic impacts directly on performance to perform actuations or sensing physical quantities, in terms of determinism and latency. The use of open-source software for real embedded systems has been described in the book (Cibrario Bertolotti, & Hu, 2015). Other examples of use of open-source software will be provided in the next sections. In the PC world, the Linux operating system is an example of widely adopted open-source platform, which offered long-term support and development. These last features are really important for systems such as CPS, which are updated and replaced infrequently. The Linux

operating system is often chosen by researchers, users and developers. Moreover, the code of the operating system, and of its software suite, is well documented, allowing to expert users a really high degree of control of any behavioral aspect. From the network point of view, Ethernet is the “de facto” standard for wired networks. Regarding wireless, a number of technologies exist, but the most common, fast and easily interoperable with Ethernet is IEEE 802.11, also known as Wi-Fi. For cost reasons and ease of configuration and installation, these technologies are the best candidates for a CPS based on open-source components. It is worth pointing out that, while the majority of protocols for industrial automation rely on Ethernet, possible candidates in industrial CPS for wireless extensions are IEEE 802.11, Bluetooth or wireless sensor networks based on the IEEE 802.15.4 standard (Lu et al., 2016).

In this chapter, a possible implementation of a CPS based on the Linux operating system is presented. The proposed architecture exploits the RTAI or XENOMAI hard real-time schedulers to guarantee the required degree of determinism of nodes. It makes use of synchronization protocols, in both the wired and the wireless extensions, to provide all the nodes a common view of time. Transmission latency can be reduced in the wired network by using hard real-time protocol stacks (such as, e.g., RTnet) and channel access methods as the time division multiple access (TDMA). Regarding Wi-Fi, redundancy techniques based on the transmission of the very same data packet on two non-overlapping networks allow to reduce both the number of packets lost and latency. Fi-

nally, we will show how the proposed architecture simplifies the integration of popular industrial protocols (e.g., EtherCAT or Modbus TCP) within the communication system.

BACKGROUND

CPSs usually interact with the surrounding environment through a wired or a wireless network. Sensing applications used to measure physical values are characterized by few timing requirements. Usually, a network wide notion of the time is maintained through specific synchronization protocols. Common time allows network nodes to associate a time to measured values. It is the case of wireless sensor networks (WSN) (Zaman, Ragab, & Abdullah, 2012) or monitoring systems. A large class of CPS applications is control systems (CSs). In CSs, remote actuators have to be continuously managed by a control unit. In its control-loop, the CS cyclically makes use of data acquired by remote sensors to properly command actuators. The main requirements of CSs are the determinism of the communication network and its reliability. In practice, a network packet has to reach the receiving node within a predefined deadline. In hard real-time (HRT) systems deadlines cannot be exceeded, while in soft real-time (SRT) systems they can be exceeded with low probability, i.e., the deadline miss ration has to be bounded. For wired networks, a number of protocols, technologies and node architectures, typically originated from the industrial world, are able to ensure HRT constraints. Nowadays, only SRT is possible for wireless networks. This is mainly due to the non-exclusive use of the communication medium (i.e., the ether) by nodes involved in the communication. Nevertheless, a number of countermeasures appeared in scientific literature to increase determinism and reliability of wireless protocols, especially for Wi-Fi. Finally, in such type of networks, nodes have to send packets with the correct timing constraints, to ensure the determinism required by the application. To this

extent, RT properties of nodes have to be improved by using, e.g., a RT operating system. In the following, design guidelines and references will be provided for the implementation of a distributed wired/wireless CPS, with completely open-source components, RT capabilities and conventional hardware.

OPEN-SOURCE ARCHITECTURE FOR A REAL-TIME CPS

Architecture

A reference network architecture of a CPS is schematically presented in Figure 1. A *wired backbone* interconnects *wired nodes*. In the proposed open-source architecture, it consists in standard Ethernet cables and switches. Some *wireless extensions* are connected to the wired backbone. They allow node mobility and cable replacement for such applications in which cabling is hard or cables are prone to wear and tear (e.g., in robotic arms). Wi-Fi has been chosen as the reference wireless technology for wireless extensions because it is faster if compared with other WSNs technologies, it is completely interoperable at the data link layer with Ethernet, and its basic components (i.e., access points (APs) and Wi-Fi adapters) are available off-the-shelf at relatively low cost. APs are used as interconnection elements between the wired backbone and wireless extensions. Two types of Wi-Fi nodes can coexist in the same CPS: conventional *Wi-Fi nodes* and redundant *Red. Wi-Fi nodes* equipped with two or more wireless adapters. The latter type of node is used for SRT applications with demanding latency and reliability requirements. Some nodes can be equipped with both wired and wireless adapters. An example is the border node of a *Wi-Fi subsystem* (e.g., the interface node of an industrial machinery produced by a specific vendor) which is connected to the wired backbone to communicate with other components of the CPS and internally exploits wireless communication for the connec-

9 more pages are available in the full version of this document, which may be purchased using the "Add to Cart" button on the publisher's webpage:

www.igi-global.com/chapter/architecture-of-an-open-source-real-time-distributed-cyber-physical-system/183836

Related Content

Computer Network Information Security and Protection Strategy Based on Big Data Environment

Min Jin (2023). *International Journal of Information Technologies and Systems Approach* (pp. 1-14).

www.irma-international.org/article/computer-network-information-security-and-protection-strategy-based-on-big-data-environment/319722

The Bees Algorithm as a Biologically Inspired Optimisation Method

D.T. Pham and M. Castellani (2015). *Encyclopedia of Information Science and Technology, Third Edition* (pp. 285-294).

www.irma-international.org/chapter/the-bees-algorithm-as-a-biologically-inspired-optimisation-method/112336

Incorporating Technology Acceptance and IS Success Frameworks into a System Dynamics Conceptual Model: A Case Study in the ERP Post-Implementation Environment

Meg Fryling (2012). *International Journal of Information Technologies and Systems Approach* (pp. 41-56).

www.irma-international.org/article/incorporating-technology-acceptance-success-frameworks/69780

Decimal Hardware Multiplier

Mário Pereira Vestias (2018). *Encyclopedia of Information Science and Technology, Fourth Edition* (pp. 4607-4618).

www.irma-international.org/chapter/decimal-hardware-multiplier/184168

Using Grounded Theory Coding Mechanisms to Analyze Case Study and Focus Group Data in the Context of Software Process Research

Rory O'Connor (2012). *Research Methodologies, Innovations and Philosophies in Software Systems Engineering and Information Systems* (pp. 256-270).

www.irma-international.org/chapter/using-grounded-theory-coding-mechanisms/63267